

# Exploring Affordances of Slack Integrations and Their Actualization Within Enterprises –Towards an Understanding of How Chatbots Create Value

Emanuel Stoeckli  
University of St.Gallen  
[emanuel.stoeckli@unisg.ch](mailto:emanuel.stoeckli@unisg.ch)

Falk Uebernickel  
University of St.Gallen  
[falk.uebernickel@unisg.ch](mailto:falk.uebernickel@unisg.ch)

Walter Brenner  
University of St.Gallen  
[walter.brenner@unisg.ch](mailto:walter.brenner@unisg.ch)

## Abstract

*The rise of chatbots poses new possibilities to link social interactions within instant messengers with third-party systems and business processes. While many companies use chatbots within the enterprise in the form of Slack apps and integrations, little is known about their affordances. Grounded in a qualitative research endeavour, we conducted 12 explorative interviews in 8 organizational settings to inductively gain rich contextual insights. Our results reveal 14 functional affordances in 4 categories, elucidating how their actualization leads to the perception of higher level affordances and constraints. First, we discuss how chatbots augment social information systems with affordances of traditional enterprise systems, and therefore, enable bottom-up automation. Second, we elaborate on how the actualization of an affordance by one user may facilitate its perception by other users. Thus, we contribute towards a better understanding of how chatbots create value.*

## 1. Introduction

Over the last few years, chatbots and conversational user interfaces have increasingly gained attention, e.g., \$200 million has been raised by startups that adopt conversational user interfaces [1]. Moreover, numerous frameworks and application programming interfaces of instant messengers facilitate the development and installation of chatbots in both, the enterprise context (e.g., Slack apps and integrations, HipChat bots, Microsoft bot framework for Skype/Microsoft Teams) and the consumer context (e.g., Facebook Messenger, Telegram Bot Platform). Besides their exploitation for the consumer-facing activities of enterprises (e.g., automating customer service), potentials to increase the productivity and efficiency of internal workflows are suggested [1].

Against this backdrop, and the increasingly dynamic and generative digital world [2], chatbots extend the functional scope of enterprise instant messengers. Con-

sequently, novel possibilities for goal-oriented action emerge for employees, i.e., functional affordances [3]. Prior empirical research that explores these possibilities of chatbots and their usage within enterprises is scarce. To date, it remains unclear how chatbots are used within organizations and how they are valuable. Consequently, the research questions at hand are as follows: *What do chatbots afford to employees and their teams within enterprises, and how can exploring their usage further advance our understanding of affordance emergence, perception and actualization?*

For the sake of clarity, we now illustrate a concrete use scenario in the context of a software developing organization [4]. Imagine that a company integrates their software deployment workflows into their instant messenger. Max, an engineer at that company, deploys software using an isolated command-line tool on his laptop. In contrast, imagine the same scenario, but with Max initiating the deployment within the instant messenger by engaging in a textual conversation with a chatbot. By shifting from a siloed terminal into a conversational group thread, the workflow becomes transparent to others. How valuable is this in everyday work?

Due to the generative nature of digital ecosystems, this is one of a potentially infinite number of chatbots [2]. This research goes beyond collecting a vast amount of use cases by investigating how chatbots are actually used for everyday work in different organizational contexts. The goal is to gain an in-depth understanding of what chatbots enable their employees to do. Therefore, we apply a functional affordance perspective (i.e., possibilities for goal-oriented actions of specified user groups [3]) to mutually explore the technological artefacts together with the capabilities and goals of employees and organizations. The focus is placed on Slack, a particularly relevant enterprise messenger with a fast growing ecosystem of Slack apps and integrations. In October of 2016, they have attracted four million daily active users with a user growth of up to three and a half over the last year [5]. Thereof over one million users in thirty-three thousand teams operate on paid accounts [5] and around

90% of these paid teams actively use apps and integrations [6]. Consequently, Slack apps and integrations are an appropriate spawn point to investigate chatbots in the enterprise context.

## 2. Background

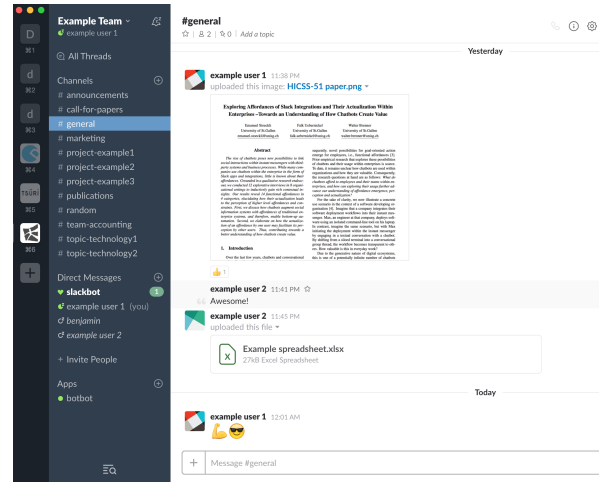
### 2.1. Chatbots

Prior research has revealed chatbots in various settings (e.g., customer support [7]), health [8]), where the common denominator is the conversation driven nature. Hence, machines that engage in conversations (i.e., conversational agents) with humans are not a novel phenomenon, per se. In fact, the well-known program ELIZA replied to human input based on keyword recognition over fifty years ago [9]. Over the years, the capabilities of information technology, particularly artificial intelligence and machine learning, have advanced. Natural language processing capabilities can now be used to extract meaning from textual input and to form reasonable responses. Thus, conversations between humans and machines are no longer limited to chatbots operating on fixed rule-based pattern matching and simple decision trees [10]. Chatbots may become artificially intelligent chat agents [8]. Much research strives to optimize such human-like behaviour, e.g., increasing the perceived humanness and engagement of conversational agents through adaptive responses [10], advancing the knowledge on human-like chatbot conversations [8].

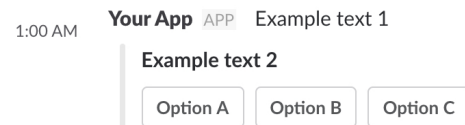
While research within the enterprise context is scarce, emerging enterprise messengers (e.g., Slack, Skype, HipChat) indicate that chatbots are by no means limited to but include chatbots that strive to act as human-like as possible [11]. Consider the illustrative case from the introduction section, where chatbot initiates a deployment workflow triggered by commands within a textual conversation. As such, the goal does not necessarily lie in increasing humanness of chatbots. Accordingly, we regard chatbots as conversation-driven integrations of third-party systems, tools and scripts into threads of instant messengers, which may occur unilaterally (e.g., posting messages to a textual thread) or bilaterally (e.g., engaging in conversations with humans).

### 2.2. Social information systems

Enterprise instant messengers can be viewed from the perspective of social information systems that follow the principles of egalitarianism, social production and weak ties [12]. We now illustrate how they apply to the widely used enterprise messenger Slack [13].



(a) Thread of a group channel within the Slack desktop app [13]



(b) Interactive buttons posted by a Slack chatbot app [14]

**Figure 1. Slack features**

*Egalitarianism* is manifested by allowing each user to create and join channels as well as contribute to and consume content. Here, a channel is a chat room visualized as a conversational thread of messages, including textual messages, images and files (Figure 1a). The channel can be set as open for users to join, or privately shared per invitation only. As such, the creation of channels and content is a result of *social production* by the users of a corresponding Slack team. Due to malleability, channels can be harnessed for various purposes (e.g., to organize conversations for specific topics of interest, technologies, individual projects but also as dedicated team channels). In turn, the visibility of such open channels enables the creation of *weak ties*. Furthermore, users can react to any kind of message that is posted with smileys (emojis), add answers directly in the main thread or create replies in a corresponding sub-thread. They can send direct messages to one or multiple people and can engage in video calls. Further emphasis is put on the search functionality offered across any type of message and the drag and drop of files [13].

On top of this, Slack offers a powerful application programming interface (API) that allows developers to build conversational driven apps and integrations into these channels, i.e., what we refer to as chatbots (Section 2.1). More specifically, by registering bot identities, developers can post messages into channels as chatbots.

They can register commands to call third-party systems and establish bidirectional connections to send and receive messages of channels. The base functionality such as reacting with emojis is also applicable to messages posted by chatbots. In addition to textual messages, Slack provides visual elements for chatbots, e.g., buttons (Figure 1b) and menus with selectable options.

Finally, Slack allows to share apps and integrations in the publicly available App Directory [11]. This has led to a growing ecosystem in terms of the number of users and the number of apps and integrations. More specifically, the App Directory listed 150 apps in December 2015 and over 385 apps in April 2016 [6]. In October 2016, we crawled already 722 apps (Section 3.1). With this ecosystem, Slack differs from previous forms of enterprise messengers that integrated chatbots (e.g., scripts integrated into Internet Relay Chat). In particular, chatbots are offered to a broader audience of users (including non-developers), which may add apps and integrations from the public directory to their channels (i.e., putting chatbots into their working environment).

### 2.3. Theory of affordances

Anchored in the objective to explore what chatbots afford to employees and their teams, we ground this research in the theory of affordances. Having its origins in the field of ecological psychology, it is guided by the logic that animals perceive what possibilities objects in their environment offer to them [17]. As widely adopted in IS [3][15][18][19][20], we build on the concept of functional affordances, i.e., "possibilities for goal-oriented action afforded to specified user groups by technical objects" [3](p. 622). Hence, this relational concept (Figure 2) has to be seen in the context of (1) a user with its abilities and goals and (2) the material properties (e.g., features) offered by the IT artefact [18][16].

There are two primary reasons why this theoretical lens suits particularly well to pursue this explorative research. First, taking up an affordance perspective allows us to mutually investigate: (1) the causal potentials of Slack apps and integrations, and (2) the goals, motivations, characteristics and capabilities of the considered employees. Second, the relational nature of affordances is fruitful for shedding light on the conditions under which affordance emergence, perception and ac-

tualization takes place for different user groups. Affordances are real, that is, they exist independently of the users perception [17]. Prior research differs between the emergence of action potentials (i.e., the existence of an affordance for a specified user), their recognition (i.e., the perception by the user) and their realization (i.e., the actualization by the user that may lead to certain effects) [16][21][15] (Figure 2). Thereby, the existing possibilities for action that an IT artefact offers to specific users are neither infinite, nor always enabling [18]. In fact, the offered possibilities may also be constraining to particular users depending on their abilities and goals [18]. The archetypes of perceived functional affordances can be distinguished along two dimensions [19]. First, the perception changes with the goals and desires of the considered user. Second, users may perceive themselves, or technology, as an actor. Affordance perception is influenced by many factors (e.g., available information) and includes the perception of non-existent affordances [16]. Perceived affordances might be actualized, depending on a users agency, and influenced by factors such as the expected outcome or the perceived efforts to take [16]. For a given user, multiple affordances may emerge from a given IT artefact. These affordances may depend on each other (e.g., the potential of analysing data requires having realized the potential of collecting data [21][18]). These affordances can also be considered on different levels, namely, on different user group levels [22][18] (e.g., individual, group, organization), and from different IT artefact perspectives [19] (e.g., on feature and system level).

### 3. Research methodology

Our research adopts an explorative approach, because prior research on chatbots, in general, and Slack apps and integrations, in particular, is scarce. Accordingly, a qualitative empirical research design is applied with the objective of enlightening this so far unexplored phenomenon inductively with rich contextual insights [23][24]. Grounding our research in the interpretative paradigm allows us to gain a deeper understanding of the meaning that individuals assign to the phenomenon of interest [25][26]. To increase rigor, we have collected and analyzed data iteratively until a coherent picture emerged [27][25].

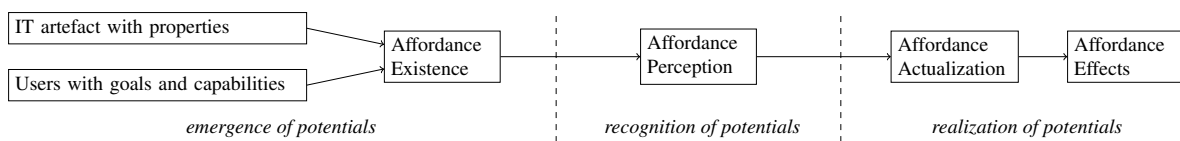


Figure 2. Affordance existence, affordance perception and affordance actualization adopted from [15][16]

### 3.1. Preliminary study

Following the call to take the IT artefact as serious as its potential effects [28], we started with a preliminary study to explore Slack and its apps and integrations. This involved examining the documentation of Slack and its API to gain an understanding of the technical capabilities. We then built a crawler, which we used to parse the Slack App Directory (Slack, 2016), on October 11, 2016 to collect the names, descriptions and the associated categories (one or more from 17 categories) for each of the available apps and integrations. In total, we collected 722 Slack apps and integrations; most of them were assigned to the following categories: productivity (n=223), social and fun (n=188), communication (n=157), developer tools (n=146) and bots (n=132). After an initial exploration of each category, we calculated association rules to determine how the categories relate to each other. From that, we obtained an understanding of the potential user groups relevant for data collection.

### 3.2. Data collection

We conducted twelve semi-structured interviews with employees from eight companies to investigate their usage of Slack apps and integrations within their work context (Table 1). The interviews lasted between 30 and 60 minutes. All interviews were recorded and transcribed right after conduction, during the period of May 2016 to January 2017. Grounded in open-ended questions, the interviewees were initially asked to describe how Slack and Slack apps and integrations are embedded in their everyday work. A further characterization of the studied enterprises is reflected in Table 1, as assimilation of Slack in terms of acceptance (i.e., committed to use), routinisation (i.e., frequently used) and infusion (i.e., comprehensive and sophisticated use) [29].

Due to the semi-structured nature, we were able to dig deeper when the interviewees mentioned interesting and unexpected ways to harness Slack integrations for their routines [23]. Further questions ranged from today's perceived action possibilities, benefits and disadvantages of Slack integrations to planned and expected future use potentials of chatbot to accomplish individ-

ual and organizational goals. From four contextual settings, at least one interviewee additionally showed us on a tablet or computer how their team uses Slack.

We build upon the empirical data from multiple social and organizational contexts, because our objective was to explore a broad range of functional affordances, which are, per se, contextual. Informed by the results of the preliminary study, we selected our interviewees as follows. First, we selected people from our network and asked the initial interviewees to recommend additional interviewees. We only included interviewees that have used both Slack and Slack integrations. The selected organizational contexts were chosen so that a variety of industries and company sizes were included. Second, the preliminary study revealed that a large share of available integrations is targeted at software developers. To prevent overemphasis on affordances for software developers, we selected an equal number of interviewees with and without programming skills (Table 1).

### 3.3. Data analysis

We conducted a qualitative data analysis consisting of iterative open, axial and selective coding [27]. More specifically, during the axial coding procedure, we considered categories of our theoretical underpinning (Figure 2). To do so, we triangulated multiple sources of evidence with the software MAXQDA 12 [24]. This included the transcribed interview recordings, notes and observations, with supplementary data provided by the companies: (1) screenshots of Slack integrations in use, (2) lists of Slack integrations in use, (3) documentations, (4) blog articles referred by one of the interviewees, which is the author, and (5) a video recording of a practice-oriented conference presentation (team of Company Beta) on how they harness chatbots for software development and operations. This allowed us to examine Slack integrations in use from different sides and in different embedded contexts [25][28]. Additionally, it was an advantage that the authors of this article use Slack integrations in their daily lives, because it helped them to understand the statements made by the participants, which is crucial for interpretative research that relies on human interpretations and meanings [26].

**Table 1. List of interview partners and context of Slack apps and integrations usage**

Id	Job position of interviewee (programming skills)	Context of Slack apps and integrations usage	Assimilation of Slack
#1	Lead Software Engineer (Yes)	Alpha (Internet of Things software provider for facility management, 5-10 employees, Switzerland)	routinisation
#2	Chief Executive Officer (No)		
#3	R&D Team Lead, Vice President (Yes)	Beta (Innovation team of a financial service provider, 1000-5000 employees, United States)	infusion
#4	Innovation Architect (Yes)		
#5	Senior Consultant (No)	Gamma (Innovation consultancy, 10-50 employees, Germany and Switzerland)	acceptance
#6	Senior Consultant (No)		
#7	Head of Development and Interaction Design (Yes)	Delta (Technical consultancy and software company, 50-100 employees, Switzerland)	routinisation
#8	Deputy Chief Technology Officer (Yes)		
#9	Product Manager (No)	Epsilon (Telecommunication, 50-100 employees within the Slack team/10k-100k in total, Switzerland)	routinisation
#10	Chief Executive Officer (No)	Zeta (Human resources & recruiting, 5-10 employees, Germany)	acceptance
#11	Head of Software and Infrastructure (Yes)	Eta (Software company, 10-50 employees, Switzerland)	infusion
#12	Innovation Manager (No)	Theta (Energy sector, 100-200 employees, Switzerland)	routinisation

## 4. Results

Embedded in eight social and organizational contexts, our research reveals 14 functional affordances of Slack apps and integrations distributed among four categories, which are elaborated upon in the following subsections. Each is illustrated by providing exemplary evidence from actualization contexts and complemented with contextual details of perceived higher level affordances and constraints.

### 4.1. Receiving messages

Four functional affordances (FA1.1 to FA1.4 in Table 2) relate to information from third-party systems and the outcomes of automated workflows posted to channels.

**4.1.1. Higher level affordances.** Being able to integrate messages from multiple chatbots and humans in one place affords to consolidate and unify communication flows. "I know that all the important things that I have to be aware of end up [in the corresponding Slack channel]" (Interviewee #11). This yields "a tremendous saving of time. Instead of gathering information up from emails and hundreds of third-party systems, I have everything at a glance" (Interviewee #11). "I see everything that happened over night, this is cool because conversations are sorted by project [channel] and I can go through by priority" (Interviewee #4). Furthermore, it relieves employees from "switching between monitors and programs all the time" (Interviewee #12). On the other hand, "it redeems from informing someone that you have pushed [source code], because it happens automatically" (Interviewee #1). Thus affording to automate information flows and, consequently, to uncouple information flows from individual employees.

In addition, our interviewees expect that actualizing FA1.1 to FA1.4 can collectively lead to increased ambient awareness. Chatbots "help team members to be more aware of the load they are putting on others" (Interviewee #3). An interviewee perceived that receiving build and continuous integration updates brings business and technology closer together. "I see where we stand, [...], what the current version contains, and new features that we have developed [relevant] for sales - so, it certainly creates proximity to the software development team" (Interviewee #2). Also within teams, "I notice what is going on in other repositories [from my team] where I'm not directly involved" (Interviewee #1).

Moreover, teams created dedicated channels, e.g., a channel for major errors, where "I know exactly, if something goes wrong, then, we really have a major problem which absolutely has to be resolved" (Interviewee #11).

**Table 2. Functional affordances related to receiving information and outcomes of automated workflows**

<i>Functional Affordance (FA)</i>	
Actualization Context	Immediate Outcome
<b><i>FA1.1 Receiving status notifications and updates</i></b>	
In software developing teams, chatbots were used to post status updates from version control (e.g., Git), issue tracking (e.g., Jira), build and continuous integration systems (e.g., TeamCity). (Alpha, Beta, Delta, Epsilon, Eta)	Status update (e.g., source code was committed, issue was resolved, new version was deployed) is automatically posted to a team or dedicated separate channel
Outgoing social media posts are officially posted by the responsible employees but relevant for all. (Alpha)	Outgoing social media communication is posted to a general channel
<b><i>FA1.2 Receiving real-time information</i></b>	
Error messages of different systems are often spread across various log files, which hinders debugging. (Eta)	Major error messages are passed to a channel to debug in chronological order.
To better understand individual test users, chatbots were used to trace a customer along a buying process. (Theta)	Each step of a customers' journey is posted to a dedicated Slack channel
<b><i>FA1.3 Receiving aggregated information</i></b>	
Fetching information from various sources (e.g., Kanban board) and preparing a structured meeting agenda for status meetings is a repetitive task that is done by a chatbot. (Beta)	Automatically generated meeting agenda is posted to project channel
Gathering lunch menus of nearby cafeterias is done by a chatbot. (Alpha)	Automatically fetched meals are posted to channel
<b><i>FA1.4 Receiving metrics and key performance indicators</i></b>	
Velocity measure as amount of functionality scrum teams deliver is repeatedly calculated by a chatbot (Beta)	An overview of the relative movement of each project is posted to a channel.
Product and service related analytics data and reviews/ratings from app stores are fetched by chatbots (Epsilon, Eta and Theta)	This feedback to the own product or service is posted to Slack (either immediately or aggregated)

As described by the interviewees, this nicely coalesces with the possibility to react to any posted content (e.g., with emojis, replies). For the interviewees, this is how it affords them facilitating discussions, as well as faster and shorter feedback cycles, which "is key in today's world. When a developer gets aware of what user x thinks about our product at the same time as I do, then, the communication simply flows much faster. And this is something we cannot achieve differently, even if I would tell it in every daily [Scrum meeting], which would be completely inefficient" (Interviewee #9). Here, the affordance to uncouple such information flows from individual employees was described with positive side effects, namely, "it also feels better [when the information of bad product reviews comes from the bot instead of the boss]" (Interviewee #9).

Finally, posting metrics and key performance indicators makes undisciplined behaviour and violations against reference values visible. This enables to enforce discipline and compliance, because "they could see the problems" (Interviewee #3). Accordingly, possibilities to shape the organizational culture emerge.

**4.1.2. Constraints.** By having machines post messages to channels, a risk of producing information overload exists. It "is basically a question of [applying the] push or pull principle... do I really have to know when someone builds or do I look it up when I need it?" (Interviewee #8). Therefore, integrating chatbots that post messages has to be well thought out, to prevent the messages from becoming constraining. In fact, "messages regarding the build processes were only interesting in 20% of the time" (Interviewee #8). Our results indicate that the way incoming chatbot messages are perceived (e.g., as an affordance or a constraint) strongly depends on the individual workflow and current context, which is not necessarily aligned with the other subscribers of a channel. Project status meeting agendas generated from metadata are perceived as "semi-interesting because I spend at least some hours on that project every day, but for people that are less involved it is very useful to run a meeting" (Interviewee #4). "These are my daily working tools, I don't need an integration that tells me hey your project has reached 50% of the billable hours, because [as a manager] I'm using this software every day" (Interviewee #8). This plays well together with the possibility of Slack to mute individual channels. Therefore, "we have thought about creating a separate channel for our chatbot and if it annoys you, then you can simply leave or mute the channel" (Interviewee #1). However, "if a separate channel is used, then, you risk not being able to reach all team members anymore. It has to be balanced how important it is that everyone sees it" (Interviewee #1). "If the channel is relatively important, [...], then you cannot mute it" (Interviewee #4). Often-times, "it depends on your job role, personally I have muted many of the channels in which chatbots regularly post messages except for the project [channels] where I'm responsible for. I have two projects where I as the architect have the technical responsibility and do have to know what my software developers commit and what bug reports are created" (Interviewee #4). Interviewees expect to be able to "create some sort of digest, which means that I would be able to choose getting information [i.e., chatbot messages for a certain channel] only once per day, week, only during nights or just if it matches certain filter criteria" (Interviewee #11).

Some interviewees perceived incoming information to be redundant, and thus, distracting. "At a certain point I had to say okay if I get this notification also here [within Slack] and I have already seen it before [notifications of the third-party system on mobile and desktop], then it distracts more than it is useful" (Interviewee #2).

Furthermore, the textual representation of information was perceived as a constraint. "I like having information graphically prepared" (Interviewee #11).

## 4.2. Getting and setting triggers

Three functional affordances (FA2.1 to FA2.3 in Table 3) ground in potentials to set and get triggers.

**4.2.1. Higher level affordances.** Setting reminders was perceived as affording automation of repetitive message flows and, thus, to relieve employees from repetitive work. "So that not every day someone has to write [to remind of the daily scrum meeting]" (Interviewee #2). Also, getting reminded to perform some task together with the potential to resolve it, was perceived as a chance to increase efficiency. "If I get reminded every evening [to capture the hours I worked], then everything is done in 3 seconds" (Interviewee #6). "I see a lot of potential for activities which are made regularly, but not every day" (Interviewee #5).

Furthermore, it was perceived as enforcing team discipline and compliance. As such, it provides "advice in accordance with our development culture" (Interviewee #3). For example, it includes problem areas and undisciplined behaviour, such as unassigned pull requests or unassigned issues. Thus, it affords to shape the organizational culture.

Interviewee #4 stated that "sometimes something goes forgotten and then you get daily a reminder" and has compared it to previous situations in which "a project manager had this [...] job to run after all to say why are there no labels and why is this like that". On the one hand, such chatbots are perceived as tools to relieve managers from repetitive micromanagement tasks, which are then executed automatically in a consistent and exact way. "It is mainly a tool that helps our man-

**Table 3. Functional affordances related to getting and setting triggers and reminders**

<i>Functional Affordance (FA)</i>	
Actualization Context	Immediate Outcome
<b><i>FA2.1 Getting reminded</i></b>	
The default Slack Bot was used to prevent from forgetting to answer an incoming message (Beta, Theta).	Reminders to answer are posted to Slack channel at a chosen time.
Chatbots were used to avoid that community tasks such as cleaning the office (Delta) or periodic tasks such as daily scrum meetings (Alpha) are forgotten.	Reminders of expectable events are posted to Slack channel.
<b><i>FA2.2 Getting nudges</i></b>	
Monitoring potential problems areas (e.g., unassigned pull requests, unresolved issues) is often annoying for managers and is therefore done by a chatbot. (Beta)	Nudging everyone to improve by creating and posting a daily digest with commitments and problem areas until they are fixed.
<b><i>FA2.3 Getting a nudge to action and resolve it</i></b>	
Time tracking is a common task of employees. However, it often gets forgotten and is difficult to do later. To prevent this, a chatbot is used (Gamma)	Reminders to capture working hours are posted and can be resolved by responding to the chatbot

agers to enforce discipline. For me as a developer it is much less useful than for him” (Interviewee #4). On the other hand, the technology is perceived as an actor that “keeps its eyes on us and barks when it saw a slip in discipline” (Interviewee #3). However, “as soon as it becomes quantitative, employees begin to play out the whole thing. If you tell me one gets points for each movement on the [Kanban] board, then, I can create micro issues and move them through” (Interviewee #4).

**4.2.2. Constraints.** On the one hand, certain teams “want to keep up the degree of freedom so that we do not really want to squeeze ourselves into processes” (Interviewee #12). On the other hand, many bot-specific constraints were perceived, such as a lack of usability. The chatbot “only asks, I give the input and then it disappears. It should also be visually recognizable what I have entered. [...] But basically I think it is good when you are able to directly resolve things” (Interviewee #5). As a consequence, other IT artefacts often afford similar goal-oriented actions, e.g., “my calendar reminds me as well with a popup” (Interviewee #5).

Furthermore, interviewees expected “an official communication to everyone or for a certain group that informs what we want to achieve, and then that applies to all. [...] And that did not exist in this case, there was no communication. Suddenly, there was a bot who has asked me now and then what I do” (Interviewee #5).

### 4.3. Queries and invocations

An additional five functional affordances (FA3.1 to FA3.5 in Table 4) are based on the potential to query and invoke the functionality of third-party systems.

**4.3.1. Higher level affordances.** Being able to capture data into third-party systems directly from Slack was perceived as valuable to achieve everything in one place, but not necessarily act as a substitute for the third-party systems. “We do not want to replace it, we still open Trello but we have the possibility to capture tasks [directly within Slack]. This makes it much easier, because you can mark something, copy, and quickly pass it over [to Trello]” (Interviewee #12). Accordingly, it relieves employees from application switching.

Moreover, being able to query and invoke functionality from third-party systems was perceived as a possibility for rapidly building prototypes without having to develop and introduce yet another employee-facing user interface. It is useful “for new features that we develop and try out. So you can prototype faster” (Interviewee #4). It was also used to provide a broader audience with access to third-party systems so “that they have an in-

**Table 4. Functional affordances related to queries and invocations within Slack channels**

<i>Functional Affordance (FA)</i>	
Actualization Context	Immediate Outcome
<b>FA3.1 Capturing data</b>	
When using Kanban boards for project management (e.g., Trello), chatbots help to capture data. (Epsilon, Theta)	Available command to add cards to Trello boards within Slack channels.
<b>FA3.2 Querying information</b>	
Querying databases requires corresponding skills. To rapidly enable employees from sales to execute predefined queries, chatbots were used (Eta)	Available commands for everyone within the actualized channel to query the latest data.
<b>FA3.3 Invoking functionality</b>	
Status reports with repetitive elements can be partially generated by a chatbot to limit the required human input to writing a summary. (Beta)	Available commands to partially generate status reports
<b>FA3.4 Invoking functionality and making invocation visible</b>	
Instead of creating video conference (e.g., join.me, appear.in) followed by sharing access details, bots are used to achieve both in one step. (Alpha, Delta)	Available commands to create and share access to a video conference within the corresponding channel.
Instead of initiating software deployments from the isolated console, chatbots are harnessed to initiate deployments within Slack channels (e.g., the corresponding project channel). (Beta)	Available commands to initiate deployment within Slack channels, which makes it visible and traceable.
<b>FA3.5 Adding gatekeepers</b>	
Due to separation of duties, developers were separated from production environments with sensitive data and glued back together using chatbots. (Beta)	Authorized legal responsibilities use chatbots to confirm and trigger the provisioning of customer environments.

terface to pull information easily [from the database]” (Interviewee #11).

Using commands (or natural language) to invoke functionality within Slack does not only afford the accomplishment of the invocation itself, it also simultaneously affords visibility to all members of the team channel. For example, when directly deploying from the project channel within Slack, then, the information about who is deploying when is disclosed. “It saves to say I now have pushed a new version, because it happens automatically” (Interviewee #1). Hence, it affords automation and consequently ensures certain information flows.

At the same time, invocations become traceable and searchable in real-time, thus, facilitating ambient awareness within the teams. “Usually just before 7, I open Slack and see what my colleagues in India have already done” (Interviewee #4). In turn, reactions and discussions enable fast feedback cycles.

Furthermore, possibilities to introduce gatekeepers arise, e.g., the default Slack bot tracks the status of people and intervenes when someone sends a message to ask if Slack should really push a notification, even though the user status is set on “do not disturb”. This is helpful “in general to reduce noise, like an personal



assistant as a gatekeeper” (Interviewee #4). In addition, invocations might be verified, delayed or prevented. In fact, chatbots afford to separate different organizational units (e.g., developers from the production environments of customers) and glue them back together.

**4.3.2. Constraints.** In the long run, querying information within Slack was found to be constraining, as compared to dedicated tools. ”The company has grown now. For the beginning it was relatively simple and also easily possible [for the sales team to query data within Slack], but the more customers and the bigger the customers, you need proper tools for data processing, aggregation and evaluation” (Interviewee #11).

Passing parameters with commands to invoke functionality was also perceived as constraining. ”We then started to build our own tools, because it is annoying to pass numerous parameters with commands and they [i.e., existing Slack apps and integrations] are often not really configurable” (Interviewee #11).

Non-technical interviewees often expressed difficulties to obtain an idea about how certain Slack apps and integrations may create value. ”What is missing are cases that point out how teams work with this and that and thus showing how it [i.e., the Slack app and integration] creates added value” (Interviewee #2).

#### 4.4. Enriching messages

Two functional affordances (FA4.1 and FA4.2 in Table 5) relate to getting outgoing messages enriched.

**4.4.1. Higher level affordances.** In the majority of the considered organizational contexts, variations of chatbots enrich messages with appropriate gif images to increase the expressive power of textual communication. Using them was found to afford fostering team cohesion, spirit and fun, because ”especially in a distributed team, it is relatively difficult to promote interpersonal commu-

nication” (Interviewee #4). Assuming that ”you have forgotten something, then, a gif is shown from Barack Obama where he drops his microphone. This is just to promote interpersonal communication and to not having quite such a dry business context” (Interviewee #4). It is ”just for fun but I think to a relatively large extent, it helps to keep the people together” (Interviewee #11)

Having messages processed and enriched with additional information, was found to play well together with the basic functionality of Slack, e.g., by automatically importing files posted as links within a message to enable searches on it. Access to files can be simplified by automatically sharing it with channel members. Overall, it affords linking and consolidating third-party systems within Slack, e.g., traditional enterprise systems such as customer relationship management systems.

**4.4.2. Constraints.** It was stated that ”much is about playing a bit and a bit of fun but what is the real value added?” (Interviewee #2). Accordingly, the use has decreased. ”When we introduced Slack, it almost exploded for three days, because for every word, you found [someone posting] a gif, which has led to whole conversations composed of GIFs, but this is not useful, just funny, what in turn is fine as well” (Interviewee #7).

## 5. Discussion

The results reveal that chatbots can be harnessed to augment enterprise messengers with affordances of traditional enterprise systems. More specifically, the identified functional affordances can be considered (1) to facilitate alignment by integrating information (e.g., receiving messages), (2) to provide control mechanisms (e.g., getting and setting triggers), (3) to enable interoperability (e.g., querying and invoking functionality of third-party systems), and (4) to increase efficiency (e.g., enriching messages) [30]. This bears similarities to prior research that has shown traditional enterprise systems that are enhanced with social features (e.g., business process management systems [12]). The phenomenon at hand shows social information systems at the focal point, which then become augmented. As such, the principle of social production goes beyond traditional user-generated content and includes the co-creation of automated information and workflows within shared conversational threads. By following the principle of egalitarianism, each member can add and use chatbots in their conversational threads. Hence, chatbots can be seen as a powerful instrument that enables organizational automation from bottom-up (rather than top-down).

Interpreting the results through the affordance theory reveals further characteristics of chatbots in general.

**Table 5. Functional affordances related to the enrichment of messages**

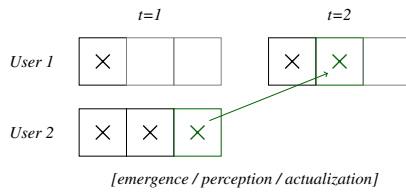
<i>Functional Affordance (FA)</i>	
Actualization Context	Immediate Outcome
<b><i>FA4.1 Having messages processed and enriched with content</i></b>	
Teams working with file management services (e.g., Dropbox, Google Drive) use apps and integrations to extend the functional scope. (Alpha, Delta, Theta)	Posted messages with links to files are processed (e.g., imported) and enriched (e.g., meta-information).
<b><i>FA4.2 Having messages processed and replaced</i></b>	
As a response to the limiting interpersonal communication with textual messages, chatbots such as Giphy were used to increase the expressive power of textual messages. (Beta, Delta, Eta)	Available command can be used together with a keyword to replace the message with a topic-related gif image.



**Affordance emergence.** The API's of instant messengers may afford technical users to develop and install custom chatbots. For most users this affordance does not exist. In turn, other users are afforded to install publicly available chatbots (e.g., from the Slack marketplace) and to add them to a conversational thread of their choice (e.g., a group channel shared with co-workers). In both cases, adding a chatbot to a shared channel leads to the emergence of new affordances for the other members of this channel. Thus, the actualization of an individual affordance of one user (e.g., to develop a chatbot) may lead to the emergence of novel shared affordances for other users (to exploit the chatbot).

**From affordance emergence to perception.** The perception of affordances depends on the available information about affordances [16] (e.g., the symbolic expressions communicated by an object to a user to reveal meaning and potential use [3]). This is a major disadvantage of chatbots, because their symbolic expressions are rare. First, the information on available chatbots is hidden and has to be polled by the user. Second, for a given chatbot, our interviewees indicated struggles with recognizing the available possibilities for action.

However, our results reveal various functional affordances of chatbots, with the characteristics that their actualization within a shared conversational thread is disclosed to others members of the channel. Therewith, the information about the availability of the actualized affordance becomes visible to others. From an affordance point of view, this indicates that the affordance actualization of User 1 can lead to the affordance perception of User 2 (Figure 3). This is where we see the biggest



**Figure 3. Actualization of an affordance by one user may facilitate its perception by another user**

value (e.g., by facilitating reactions, discussions, ambient awareness) and likewise the biggest challenge arising from chatbots that are integrated in shared conversational threads. The channel in which a chatbot is added to and in which the corresponding affordances are eventually actualized is socially determined by the actualizing user. Thus, the same technical object can be put into different contexts by the appropriating user and, thus, lead to different kinds of affordances (e.g., individualized or shared affordances [22]) or constraints.

**From affordance perception to actualization.** Prior research suggests that the actualization and, ac-

cordingly, the realization of effects, is influenced by the actualization effort and the expected outcome [16]. However, the perceived expected outcome depends on how the other members of the channel act. In fact, our results reveal constraints, such as information overload. Affordances have to be seen in the context of the alternatives, because employees constantly compare affordances [21]. Thus, the actualization effort has to be balanced with the perceived expected outcome. Future research should investigate how to reduce the effort needed to actualize the affordances of chatbots. We see the moves of platforms, such as Slack, to exploit buttons and selectable menus as a starting point. Furthermore, chatbots have to be designed to reduce the widely perceived constraints, such as information overload.

## 6. Conclusions and limitations

In summary, the contributions of this research are threefold. First, we contribute to the literature on social information systems by shedding light on the previously unexplored and novel phenomenon of chatbots in the context of enterprise messengers. We do this by pointing out 14 functional affordances from four categories: receiving messages, getting and setting triggers, executing queries and invocations, and enriching messages. Emergent from the actualization journeys, we elaborated on higher level affordances, such as consolidating communication (e.g., have messages from multiple people and third-party systems in a chronologic thread) and automating information and work flows (e.g., ensure discipline, relieve employees, shorten feedback cycles, shape organizational culture) and constraints. Second, we contribute to the body of affordance literature by discussing how the actualization of an affordance by one user may affects the affordance perception of other users (Figure 3), whereas prior research has shown dependencies between individual affordances [21] and different levels of affordances, e.g., individual and organizational [18]. Third, the paper at hand informs practitioners about functional affordances as well as constraints of Slack apps and integrations within enterprises.

Nevertheless, our findings are subject to limitations. Due to the qualitative and interpretive research design, exhaustiveness cannot be guaranteed. Moreover, our research focused on Slack in different organizational contexts. Future research should consider further enterprise platforms (e.g., Skype) and organizational contexts.

## 7. References

- [1] D. Schatsky and P. Gratzke, "The conversational enterprise - how chatbots can answer questions for both cus-

- tomers and employees,” tech. rep., Deloitte. University Press, 2016.
- [2] A. Eck and F. Uebernickel, “Untangling generativity: Two perspectives on unanticipated change produced by diverse actors,” *ECIS 2016 Proceedings*, 2016.
  - [3] M. L. Markus and M. S. Silver, “A foundation for the study of IT effects: A new look at DeSanctis and poole’s concepts of structural features and spirit,” *Journal of the Association for Information Systems*, vol. 9, no. 10/11, p. 609, 2008.
  - [4] O. Vehviläinen, “ChatOps: Everything about deployments right inside your chat.” <http://blog.flowdock.com/2014/11/11/chatops-devops-with-hubot/>, 11 Nov. 2014. Accessed: 2016-11-30.
  - [5] “Bots, the next frontier.” <http://www.economist.com/news/business-and-finance/21696477-market-apps-maturing-now-one-text-based-services-or-chatbots-looks-poised>, 2016. Accessed: 2017-5-30.
  - [6] Slack Platform Blog, “Investing in the future of work.” <https://medium.com/slack-developer-blog/investing-in-the-future-of-work-d4ffb87f985#1fns4wife>, 19 July 2016. Accessed: 2016-11-30.
  - [7] M. Lasek and S. Jessa, “Chatbots for customer service on hotels websites,” *Information Systems in Management*, vol. 2, no. 2, pp. 146–158, 2013.
  - [8] R. Crutzen, G.-J. Y. Peters, S. D. Portugal, E. M. Fisser, and J. J. Grolleman, “An artificially intelligent chat agent that answers adolescents’ questions related to sex, drugs, and alcohol: an exploratory study,” *The Journal of adolescent health: official publication of the Society for Adolescent Medicine*, vol. 48, pp. 514–519, May 2011.
  - [9] J. Weizenbaum, “ELIZA—a computer program for the study of natural language communication between man and machine,” *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.
  - [10] R. Schuetzler, M. Grimes, J. Giboney, and J. Buckman, “Facilitating natural conversational agent interactions: Lessons from a deception experiment,” in *ICIS 2014 Proceedings*, 2014.
  - [11] Slack, “App directory.” <https://slack.com/apps>, 2016. Accessed: 2016-11-30.
  - [12] R. Schmidt and S. Nurcan, “Augmenting BPM with social software,” in *Business Process Management Workshops*, Lecture Notes in Business Information Processing, pp. 201–206, Springer, Berlin, Heidelberg, 2009.
  - [13] Slack, “Where work happens.” <https://slack.com/is>. Accessed: 2017-6-1.
  - [14] Slack, “Slack API.” <https://api.slack.com/>. Accessed: 2017-6-1.
  - [15] G. Pozzi, F. Pigni, and C. Vitari, “Affordance theory in the IS discipline: a review and synthesis of the literature,” in *AMCIS 2014 Proceedings*, 2014.
  - [16] E. Bernhard, J. C. Recker, and A. Burton-Jones, “Understanding the actualization of affordances: A study in the process modeling context,” in *ICIS 2013 Proceedings*, eprints.qut.edu.au, 2013.
  - [17] J. J. Gibson, “The theory of affordances,” in *Perceiving, Acting and Knowing* (S. R. and Bransford, ed.), pp. 67–82, Lawrence Erlbaum Associates, 1977.
  - [18] D. M. Strong, S. A. Johnson, B. Tulu, J. Trudel, O. Volkoff, L. R. Pelletier, I. Bar-On, and L. Garber, “A theory of Organization-EHR affordance actualization,” *Journal of the Association for Information Systems*, vol. 15, pp. 53–85, Feb. 2014.
  - [19] A. Savoli and H. Barki, “Functional affordance archetypes: a new perspective for examining the impact of IT use on desirable outcomes,” in *ICIS 2013 Proceedings*, aisel.aisnet.org, 2013.
  - [20] E. Stoeckli, F. Uebernickel, and W. Brenner, “Capturing functional affordances of enterprise social software,” in *AMCIS 2017 Proceedings*, 2017.
  - [21] P. Glowalla, C. Rosenkranz, and A. Sunyaev, “Evolution of IT use: A case of business intelligence system transition,” in *ICIS 2014 Proceedings*, 2014.
  - [22] P. M. Leonardi, “When does technology use enable network change in organizations? a comparative study of feature use and shared affordances,” *MIS Quarterly*, vol. 37, no. 3, pp. 749–775, 2013.
  - [23] G. Paré, “Investigating information systems with positivist case research,” *The Communications of the Association for Information Systems*, vol. 13, no. 1, p. 57, 2004.
  - [24] R. K. Yin, *Case Study Research: Design and Methods*. Applied Social Research Methods, SAGE Publications, 2008.
  - [25] H. K. Klein and M. D. Myers, “A set of principles for conducting and evaluating interpretive field studies in information systems,” *Management Information Systems Quarterly*, vol. 23, no. 1, pp. 67–93, 1999.
  - [26] G. Walsham, “Interpretive case studies in IS research: nature and method,” *European Journal of Information Systems*, vol. 4, no. 2, pp. 74–81, 1995.
  - [27] J. M. Corbin and A. Strauss, “Grounded theory research: Procedures, canons, and evaluative criteria,” *Qualitative sociology*, vol. 13, no. 1, pp. 3–21, 1990.
  - [28] W. J. Orlikowski and C. S. Iacono, “Research commentary: Desperately seeking the “IT” in IT Research—A call to theorizing the IT artifact,” *Information Systems Research*, vol. 12, no. 2, pp. 121–134, 2001.
  - [29] X. Wang, K. Conboy, and M. Pikkarainen, “Assimilation of agile practices in use,” *Information Systems Journal*, vol. 22, pp. 435–455, 1 Nov. 2012.
  - [30] T. Mettler and R. Winter, “Are business users social? a design experiment exploring information sharing in enterprise social systems,” *Journal of Information Technology*, vol. 31, no. 2, pp. 101–114, 2016.